

REMARKS/ARGUMENTS

The Office Action of 06/27/07 has been carefully reviewed and these remarks are responsive thereto. The Examiner has entered an amendment filed April 20, 2007, the Examiner having considered a proper request for continued examination efiled on June 5, 2007. On August 30, 2007, applicants requested a personal interview with the Examiner and her supervisor to advance prosecution on the merits. As a result, a personal interview among Examiner Ahluwalia, Supervisory Primary Examiner Hosain Alam and Applicant's undersigned representative occurred with J. Douglas Birdwell, an inventor, attending by telephone. Issues identified for discussion include a submission under seal of a confidential document not labeled as confidential by the publishing party. That issue has been considered by Special Programs Examiner Vincent Trans and has been rendered moot. Mr. Trans will assist Ms. Ahluwalia to assure that the document submitted under seal is treated as confidential.

In addition, Applicants indicated that they were in a position to propose further amendments to the claims and would be discussing their proposed amendments at the interview. Applicants wish to thank the Examiners for extending the courtesy of an interview, and prosecution on the merits has been further advanced thereby.

The state of the claims as presently presented is as follows. Claims 1-41, 43 and 67 have been canceled. Claims 42, 44-46, 48-49, 58-59, 61, 64-66 and 68-75 have been amended. Consequently, claims 42, 44-66 and 68-76 remain pending. By way of example, applicants have amended independent claim 42 to clarify the claimed parallel data processing architecture for search, storage and retrieval of data of a database responsive to client queries for specific data of said database responsive to client queries for specific data of said database. As will be explained below, claims 42, 44, 45, 58, 66 and 72 have been amended to clarify "according to a time constant" per, for example, the specification support at p. 48 to relate to a recited "balance." Reconsideration and allowance of the instant application are respectfully requested. This response to the Office Action of 06/27/07 contains amendments to independent claims and clarifies dependent claims such that the claims patentably distinguish over the combination of references made by the Examiner of Merkey/Kitain as will be further explained below.

Generally, the amendments made herein merely clarify claim limitations of previously examined claims to define and clarify such terms as client query, search queue, database index, nodes of a database tree of said database, bringing into balance, capacity as “processor capacity” and load as “search queue length load” and the like which are not taught by either Merkey or Kitain. In particular, while Merkey teaches a parallel data architecture, Merkey is not a database search engine and for reasons further described herein cannot be combined and teaches away from Kitain in meeting the present claims as amended. Applicants respectfully request reconsideration and allowance due at least to the Examiner’s failure to present a *prima facie* case of obviousness of the amended claims as will be discussed further below.

In her Advisory Action, the Examiner indicates in item 2: “Applicant argues that there is no teaching in Merkey of a communication system...wherein at least two host processors communicate capacity and load information to other host processors.” “The Examiner submits that Merkey teaches the communication of the capacity and load to other processors in column 9 lines 31-50, Merkey. It discloses the communication of the load and capacity between processors. Furthermore, support for this communication is also disclosed in column 10 lines 59-67, Merkey.”

The Examiner’s argument crumbles because the Examiner must read a given claim as a whole as now amended. Merkey does not teach or suggest a parallel data processing architecture for search, storage and retrieval of data of a database responsive to client queries for specific data of said database. Merkey fails to disclose the recited communication system and associated elements such as “broadcasting” and bringing its search queue of said client queries into balance “according to a time constant.” Merkey fails to teach “load information of a search queue length of said client queries” among existent and other newly added limitations which are respectfully submitted to patentably distinguish over Merkey or a Merkey/Kitain combination. Many independent claims have been amended to define load as search queue length and capacity as processor capacity where Merkey confuses load and capacity. Merkey, for example, not comprising a search engine, fails to teach a “search queue length load” as recited.

The Examiner’s reliance on Merkey column 9, ll. 31-50 is misplaced. The entire passage, applicants’ emphasis added, reads as follows:

“FIG. 7 further illustrates the processor thread queue control structure 81 for processor P1; the control structures 81 of the other processors are organized in a similar manner. The control structure 81 includes a load indicator 86 which indicates how heavily *the corresponding processor* is loaded. That is, the load indicator 86 provides a measure indicating how much of the available processing capacity is being spent running code in application threads 66 (FIG. 6) versus how much capacity is spent running the idle thread 84, waiting for I/O to complete, or otherwise supporting the application threads 66.

A presently preferred load indicator 86 includes a sleep request count 88 that indicates how often the threads 66 running *on the corresponding processor* have been suspended in mid-execution. Lower values in the sleep request count 88 indicate busier processors, and higher values indicate idler processors. Those of skill in the art will appreciate that other measures may also be used as load indicators 86, including without limitation, cycles spent in the idle thread 70, 84.”

The Examiner misreads the quotation as disclosing or suggesting that processors P1-P4 communicate with each other when it is clear that each processor only maintains information about itself as shown, for example, in Figure 7: sleep request count, eligible thread count, lockable queue mutex and the like. If the examiner is suggesting that a “sleep request count 88” is communicated by another processor to the corresponding processor, the examiner is mistaken as the words clearly indicate otherwise.

Sleep request count is a very coarse measure of how busy a process is. Col. 9, ll. 47-50 state that load indicators include “cycles spent in the idle thread.” This may be an indication of “capacity” according to the equation: load =’s total capacity – cycles spent in the idle thread. But it is not an indication of “search queue length load” and “processor capacity” as recited. There is no communication among Merkey processors of capacity and load as recited. It is clear that Merkey teaches maintaining at a given processor “how heavily the corresponding processor is loaded” and a “sleep request count 88.” The preferred disclosed communication is not with another processor but with a “global control structure 67” (FIG. 8).

To the contrary to the Examiner's position, Merkey teaches away from applicants' "communication system" because a global control structure 67 (FIG. 8) is utilized which "includes a global dispatch queue 68 that holds threads 66 which are waiting for a processor." Merkey, consequently, fails to teach or suggest the communication system as recited.

Merkey fails to teach or suggest a search queue as recited "of client queries" because it is not a database search engine. To the extent a queue is discussed see Merkey col. 10, ll. 41-47 which indicates: "When a processor Pn becomes available, a scheduler for that processor searches the queues 62 to locate a thread 66 to run on that processor. The processor is then allocated to that thread 66. If no application threads 66 are ready to run, the search will locate one of the idle threads 70, 84. Otherwise, the first application thread 66 found will get the processor."

Read together, these passages from columns 9 and 10 of Merkey suggest that searching occurs of *thread* queues of other processors but such searching by one processor of another is not "a communication system...wherein at least two host processors communicate capacity and load information to other host processors" especially in the context of retrieval of data of a database responsive to queries for specific data of said database as recited. In the first instance, the communication system as recited requires the communication of capacity **and** load. As indicated above, load and capacity are two variables that are related but are independent of one another and, in many independent claims defined separately as processor capacity and search queue length load. Thus, the Examiner's allegations fail on at least two grounds 1) both capacity and load are not communicated in Merkey, only one or the other and 2) there is at best a "search" in Merkey but not a "search queue of said client queries" where client queries are "for specific data of said database." Merkey does not teach or suggest "a communication system...wherein at least two host processors communicate capacity and load to other host processors" as recited especially in the context of a recited "broadcast" or "broadcasting" or bringing a search queue into balance as recited "according to a time constant."

Claim 45 as amended, for example, reads "bringing its search queue of client queries into balance with another host processor according to a time constant." Claims 46 and 48 recite "at least two host processors communicate capacity and load information to other host processors;

selected host processors storing a database index for said database comprising nodes of a database tree for said database and data accessible via said nodes of said database tree.” Claim 66 recites: “each host processor broadcasting its capacity and load information to other host processors” and later in the claim “each host processor bringing its search queue of client queries into balance with another host processor according to a time constant responsive to receipt of said broadcast capacity and load information, said balancing including exchanging unprocessed search requests with a recipient host processor responsive to a stochastic selection process to determine the recipient host processor of an exchanged search request between host processors, said balancing thereby minimizing the time required to respond to client queries for retrieval of responsive data from said database.” Claim 68 has been written in independent form to include all the elements of claim 66 and explain “bringing a search queue of client queries into balance” as further comprising “exchanging a block of search requests between host processors, thereby minimizing a time required to respond to client queries by said exchanging of said block of search requests.” Claim 72 is also amended to clarify and to patentably distinguish: “said bringing into balance comprising equalizing average waiting times for service and computation between said search engines.”

An inventive aspect of many of the independent claims is a recited “balance.” Balancing a processor/system of processors is not important to Merkey: Abstract “threads tend to stay with that processor unless the system load is severely unbalanced ...”; col. 5, ll. 51-54, “moving threads 44 between processors may severely degrade system performance because it undercuts the performance gains ...”; col. 11, ll. 21-23, “threads 66 tend to stay on a given processor until the system load becomes very uneven, with some processors being very busy and others being mostly idle.”

In a system according to applicants’ disclosure, tasks are waiting to be executed, not being executed as in Merkey. Consequently, there are no cache penalties for moving threads and so, for example, “bringing its search queue into balance with another host processor according to a time constant in response to receipt of said broadcast capacity and load information” or a related balance process is recited in claims and suggested to the examiner as a distinguishing feature of applicants’ invention.

The examiner is respectfully requested to reconsider the claims and draw a line of allowable subject matter as Merkey is clearly missing recited elements as indicated above.

The burden is on the Examiner to show in Merkey where the recited “maintaining a list of available host processors,” “communication system,” “broadcasting,” “balance,” “nodes of said database tree” and “time constant” among many other recited elements are found – none of these are found at col. 9, ll. 31-50, even in combination with col. 10, ll. 41-47 and other passages of Merkey relied upon thus far by the Examiner.

Claim 58/46 has been amended to correct “according to a time constant” to be consistent with the specification and further recites “said balancing including exchanging unprocessed search requests with a recipient host processor responsive to a stochastic selection process” among other amendments such as clarifying load as “search queue length load information.” Claim 65/64/42 has been amended to be consistent with the specification as described per Example 4, page 20-23 and to recite the application of the parallel data processing architecture “wherein said database comprises DNA profiles” supporting the recitation “number of alleles” which appears in the claim and also: “said bringing its search queue into balance with another host processor according to a time constant minimizing the time required for performing a search request for DNA profiles satisfying a set of selected database search criteria.”

Applicants wish to direct the Examiner’s attention to claims 50/49/42 and 51/49/42 because neither Merkey nor Kitain teach or suggest what happens in the event of an addition of a host processor or a failure.

Applicants also wish to direct the Examiner’s attention to the feature of claim 61/42 of “executing a set of tests, associating one test to each non-terminal node of said database index for said database.”

Applicants also wish to point to limitations of claims 66 and 72 directed to “stochastic selection process to determine the recipient host processor of an exchanged search request between host processors thereby minimizing a time required to respond to client queries for retrieval of responsive data from said database” or “stochastic selection of host processors to determine a recipient host processor for an exchanged search request . . .”

Applicants reserve their right to continue prosecution of claims rejected responsive to the first Office Action issued July 26, 2006 and cancelled in any amendment in this application in a continuation application as applicants continue to traverse the rejections made in the first Office Action.

The Rejection of Claims 42-76 Under 35 U.S.C. §103(a)

Claims 42-76 are rejected as unpatentable over Merkey (U.S. 6,728,959 B1) in view of Kitain et al. (U.S. 5,864,871). The rejection is respectfully traversed.

To reject a claim as *prima facie* obvious three criteria must be met:

First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations.

MPEP § 2142. The rejection of the present claim set fails to meet at least the first and third criteria with respect to the amended independent claims (now incorporating limitations defining the database search and retrieval environment of the claimed embodiment). There is also no reasonable expectation of success of making a Merkey/Kitain combination because they will not operate with one another and teach away from one another.

Applicants disagree with the Examiner.

Applicants wish to direct the Examiner's attention to at least page 48, ll. 7-18 where support may be found for "according to a time constant" of amended independent claims. As is clear from the specification, at page 48, the time constant relates to load balancing and not broadcasting. Merkey teaches away from balancing load according to a time constant as discussed above.

The rejection of each independent claim will now be discussed and a demonstration made that the rejection of each independent claim as amended is not well-founded.

Claim 42 as amended

The Examiner in regard to claim 42 states that the limitations of the claims are found in Merkey, Figures 2, 5, 6, 7 and 8: “host and root host processors maintaining a list of available host processors and information about the capacity and load for each available host processor in memory.” Office Action at page 4, lines 1-3. Applicants disagree.

Merkey searches thread queues rather than having “a search queue of said client queries” as recited where client queries are defined as “for specific data of said database.” There are three types of thread queues in Merkey: the global dispatch queue, the unlocked local queue and the lockable local queue (which is optional), none of these being a “search queue of said client queries” as recited. A Merkey “processor in question” has exclusive control over the unlocked local queue, which therefore does not need a mutex and can move threads from the unlocked local queue to the global dispatch queue. Another processor or the “processor in question” may then select and move one of the threads from the global dispatch queue. The local thread queue is just that – a local thread queue for a given processor; a Merkey “processor in question” looks to the global search queue 68 first as will be explained below. Again, Merkey has no “search queue of said client queries” “for specific data of said database” as recited.

The recited “communications system” is allegedly shown in Figure 9. But communication is of threads, not load and capacity. The examiner admits that Merkey does not disclose “the search, storage and retrieval of data and the database index” as claimed. The Examiner relies on Kitain, column 6 lines 12-27 and column 16 lines 44-47. But Kitain does not teach the communications system or a bringing into balance as recited. Applicants cannot agree that these references disclose or suggest applicants’ claim 42 as amended.

Merkey shows in Figure 9: “place all processors on candidate list,” step 100, “remove processors with too few eligible threads,” step 102, “any processors left?,” step 104; “identify busiest remaining processor,” step 108 and so on. Applicants previously urged a difference between load and capacity in distinguishing claim 42 over Merkey, but there is more recited in amended independent claim 42 than a distinction between load and capacity. Merkey, column, 9

lines 31-40 teaches “load indicator 86” for “available processing capacity” that is maintained in processors P1 through Pn. Thus, Merkey confuses the terms load and capacity. Original and amended claim 42 recites that both load and capacity are broadcast by at least two processors to another processor and maintained for all processors at each processor (not suggested or described by Merkey’s “global thread queue control structure 67” shared by all processors).

Merkey teaches thread count as a measure of load/capacity. “Thread count for each processor is maintained,” column 12, lines 16-26. See also Figure 10 and, at column 12, lines 26-39, there is a discussion of two seconds large time quantum and small time quantum. Step 114 is a regular checking for movable threads 66 in an unlocked local queue 64 performed at every “large time quantum.” See Merkey, col. 12, ll. 16-64 wherein it is described how a processor is allocated to a thread *as opposed to plural processors communicating load and capacity information to other processors and maintaining capacity and load for each available processor as recited* (our emphasis added), i.e. “No mutex is needed for the unlocked local queues 64 because they are only accessed by the local scheduler for the processor in question” (Merkey, col. 12, ll. 59-61). Simply stated, Merkey gathers at a central site, queue control structure 67 in global dispatch queue 67. To the contrary, amended claim 42’s communication system and processor memory limitations call for communication by plural processors to other processors and having processor memory of load and capacity for other processors (not just for its own thread queue), not shown by Merkey. Merkey teaches away from maintaining other than local information at local queue 64 – a processor in question looks first to global dispatch queue 67 and then to other local queues 64 to move a thread to the global dispatch queue (Merkey, col. 11, ll. 13-20). So Merkey fails to teach at least the communication system and processor memories of amended claim 42. All the independent claims have the advantage that a given processor may independently utilize load/capacity information stored in its memory without regard to other processor activity because each host processor maintains load/capacity information for the other host processors in the parallel data processing architecture.

As earlier urged, Merkey appears to teach a load indicator that “provides a measure indicating how much of the available processing capacity is being spent running code in

application threads....” Merkey at col. 9, lines 36-38. Merkey does not appear to teach maintaining information about processor capacity as defined in the art at each processor as recited for other processors. “Processing capacity”, as defined by Microsoft Press *Computer Dictionary*, is “the maximum number of operations that a processor can handle in a given unit of time.” See Exhibit 1 at page 54, lines 28-30, left hand column. It appears that Merkey’s load indicator (threads) may provide the proportion of processing capacity of a processor that is currently in use, not the capacity of a processor as is known in the art, or “the maximum number of operations that a processor can handle in a given unit of time.” Claim 42 recites “each (of plural processors) maintaining a list of available host processors and information about the capacity and load for each available host processor.” Merkey teaches one centralized list 68 and local thread queues 64 for a given processor and so no communication system or processor memories as recited requiring multiple processors communicating the load and capacity to other processors and multiple storage of capacity and load information.

Moreover, Merkey teaches a load indicator “which indicates how heavily the **corresponding processor** is loaded.” Merkey at col. 9, lines 34-35 (emphasis added). It appears as if Merkey’s load indicator only maintains information indicating “how heavily the corresponding processor is loaded” at local queue 64, not *each* available processor, as recited in claim 42 as amended – i.e. plural processors. As mentioned above, amended claim 42 recites “each of said host and root host processors maintaining a list of available host processors and information about the capacity and load for each available host processor.” This feature is not taught by Merkey.

Amended claim 42 requires at least two host processors have a search engine and maintain information of a search queue of said client queries, “selected host processors storing a database index for said database comprising nodes of a database tree for said database and data accessible via said nodes of said database tree” and “each of said host and root host processors maintaining a list of available host processors and information about the capacity and load for each available host processor in memory,” “each of said host . . . processors bringing its search queue into balance with another host processor according to a time constant in response to

receipt of broadcast capacity and load information.” Merkey has no search engine and teaches no “search queue of said client queries” as recited “for specific data of said database.” To the contrary, Merkey centralizes operations at a “processor in question” corresponding first with a global dispatch queue control structure 67, for example, as shown in Figure 6 (col. 9, line 1). But according to amended claim 42, “at least two host processors communicate capacity and load to other host processors”, not one processor in question maintaining the load information for itself and looking to a central structure 67 first and then to other local queues. Merkey does not teach communicating to others and maintaining lists of processors and loads for other processors. Thus, Merkey fails to teach the communication system and memory limitations as recited whereby claim 42 as amended requires “at least two host processors communicate capacity and load information to other host processors” where each processor maintains load and capacity of others. To the contrary, Merkey teaches a centralized queue structure 67 and the “processor in question” gathers thread count from others after checking the central queue 67 – not plural processors broadcasting to others as recited: “Threads are moved to a different processor only after being moved from an unlocked local queue into the global dispatch queue and thence to another processor” (Merkey, col. 14, ll. 20-29).

Moreover, the Examiner admits that Merkey does not disclose “the search, storage and retrieval of data and the database index as claimed” and so relies on Kitain.

At most, Kitain appears to disclose a web server 4 coupled to at least two database servers of a central server 2 and a further central site 1 (Figure 1). Kitain fails to teach or suggest “a communication system coupling said host and root host processors,” as recited. Even if combined, the alleged combination of Merkey and Kitain would not teach or suggest the features of independent claim 42 as amended. Kitain does not make up for the two processor broadcast or memory deficiencies of Merkey (Merkey teaching only one “processor in question” trying to move a thread responsive to a thread count value) because Kitain, also has operations centralized at one of central site 1, central server 2 or web server 4 of Figure 1. Kitain also fails to teach two processors communicating to other processors as recited or memories as recited. Moreover, there is no suggestion to combine Kitain with Merkey because, like Merkey, Kitain operates

from a single central site (Figure 1) while the invention as claimed relates to multiple processor communication and storage of load and capacity.

In fact, Kitain cannot be combined with Merkey because Kitain teaches away from Merkey at column 16, lines 44-47 to the extent that neither deals with the problem of unbalance. Kitain’s “CGI practices semi-random selection of servers in an effort to balance the load on servers. This means that the order that servers are tried is not always the same.” To the contrary, claim 42 as amended requires “bringing its search queue (as defined of client queries) into balance with another host processor according to a time constant in response to receipt of said broadcast capacity and load information.” There is at least no time constant in Kitain. Merkey as urged above teaches a quantum of time but fails to teach any kind of search queue balance according to a time constant (and the examiner appears to admit Merkey’s failure to teach or suggest a search engine). One would not look to Kitain for the answer to a problem of search queue balance not even presented in Merkey. Moreover, Kitain says nothing about balance “according to a time constant” and so would not be combinable with Merkey.

In summary, then, Merkey and Kitain even if combined fail to teach at least the limitations of claim 42 as amended requiring two host processors bringing into balance their search queues according to a time constant in response to broadcast capacity and load information by at least two processors to other processors and memories maintaining other processor capacity and load information. Moreover, one would not be motivated to combine Kitain, for “search queue” and “database index” limitations of amended claim 42, and Merkey because they teach away from one another. The examiner speaks to motivation at page 5 in terms of providing “an efficient search engine.” But Merkey is not a search engine and so the examiner is using improper hindsight by referring only to Kitain as being efficient but not demonstrating any need in Merkey for Kitain’s teachings or discussing the problems raised in trying to combine the teachings of Merkey and Kitain operating from differing principles: checking thread count every time constant (Merkey) and semi-random selection of a first processor and then a second and so on for a non-text matching or a text matching query (Kitain,

col. 16, ll. 29-47). The examiner having failed to make a *prima facie* case of obviousness is respectfully requested to reconsider claim 42 as amended.

Claim 44 as amended

Claim 44 incorporates many of the clarifying limitations of claim 42 to define the database search environment, for example, “of a database responsive to client queries for specific data of said database.” Consequently, the examiner’s rejection of amended claim 44 is respectfully traversed for all the reasons above with respect to claim 42. The Examiner relies on Merkey’s Figure 6 depiction of a plurality of processors and a global thread queue control structure 67 and its global dispatch queue 68 in combination with Kitain’s col. 6, lines 24-27 to reject claim 44. Claim 44 further recites that there are three (or more) host processors, of which two have search engines and maintain information of a search queue of said client queries and the third comprises the root host processor. Admittedly Merkey shows processors P1 to Pn. However, the examiner admits that Merkey does not disclose search, storage and retrieval of data and the database index comprising nodes and data accessible via said nodes. The Examiner has rejected claim 44 by reading in an alleged suggestion of Kitain or Merkey that two host Merkey processors have search, storage and retrieval from column 6, lines 24-27 and then refers to Merkey for the third processor comprising the root host processor as per Figure 6. This reading in is improper hindsight.

Firstly, while Kitain shows “at least two database search engines” at column 6, lines 12-27, there is no suggestion raised by the examiner that one would be motivated to magically create Merkey’s processors P1 to Pn of Figure 6 to include two with search engines maintaining search queues as defined that are balanced according to a time constant on receipt of capacity and load information as recited. Moreover, there is no suggestion in Kitain to designate one processor of Merkey as a root host processor. The Examiner uses improper hindsight to simply designate one processor, rather than two with search engines as described by Kitain, as a root host. So, for at least these additional reasons, claim 44 as amended patentably distinguishes over Kitain or Merkey or their unmotivated combination.

Claim 45 as amended

Claim 45 is allowable for all the reasons that claim 42 as amended is allowable. Claim 45 as amended includes previously unsearched limitations such as “maintain load information of a search queue length of said search queries” and so on. Claim 45 is a slightly different twist in designating processors: “selected host processors storing a database index for said database comprising nodes of a database tree. . .” The examiner rejects claim 45 relying on Merkey Figure 6 in combination with Kitain. As already explained, Merkey’s global queue control structure 67 and global dispatch queue 68 is associated with all processors, P1-P4 according to Figure 6 and depends on a “processor in question”. Now, according to claim 45 as amended, the recited plurality of host processors comprises two host processors, of which one comprises the root host processor and both the processors have search engines and maintain information of a search queue balanced as recited in amended claims 42 and 43 “according to a time constant. As with claim 44, the Examiner cannot find a suggestion to convert certain of Merkey’s processors P1 to Pn into processors with search engines, one of which is the root host, nor does Merkey teach a search engine or search queue. Kitain, for example, teaches away from this because no candidate central site 1, 2 or 4 (Figure 1) is designated as a root host, neither is one of DB servers 11 or 13 of server 2 so indicated. As explained previously, an active processor tries one processor at a time in Kitain (Kitain, col. 16, ll. 29-47). So, for at least these additional reasons, claim 45 patentably distinguishes over Kitain or Merkey or their unmotivated combination.

Claim 46 as amended

Claim 46 is allowable for all the reasons that claim 42 as amended is allowable. Merkey fails to teach, for example, the recited processor memory and communication system elements, the limitations of search, storage and retrieval of data of a database responsive to client queries and a database index for said database and so on as admitted are missing by the Examiner and the failure to meet the communication system limitation of each of said at least two processors broadcasting capacity and load to other host processors for storage in processor memory. Amended claim 46 requires a root host responsive to a client query and using an initial search

queue of at least said client query where the client query is defined as “a client query for said specific data of said database.” There is required a root host processor that maintains a list of available hosts and information about capacity and load; two host processors communicating capacity and load to other host processors and including an initial search queue limitation that cannot possibly be met without bringing Kitain into the equation. The examiner rejects claim 46 based on Kitain’s allegedly teaching the root host processor initial search queue limitation per Figure 4 and column 11, lines 42-53. However, these references are silent about the recited limitation. As already explained, there is no suggestion to magically designate a processor in Merkey (or Kitain) as a root host processor per Figure 6 or to suggest that such a root host processor so magically designated is responsive to a client query and uses an initial search queue. Kitain admittedly discloses DB servers 11 and 13 of a central repository 2, a web server 4 and a central site 1 per Figure 1. Figure 4 shows the word query 122 and a display screen Query Results. The referenced passage at column 11 of Kitain, rather than describing a root host responsive to a client query, describes the network of Figure 1 wherein the central server 2 comprises servers 11 and 13 and does not describe a root host responsive to a client query and using an initial search queue. But as explained above, Kitain attempts to locate a new processor for a search query one at a time and so performs no communication system or processor memory as recited (Kitain, col. 16, ll. 29-47). The burden is on the Examiner to particularly describe where the limitation is shown or described. The Examiner cannot simply suggest that because there exists a query per Figure 4 that there is a root host responsive to a query and using an initial search queue when neither Merkey or Kitain show the communication system or processors memories as recited, let alone, that a root host processor (one of a plurality of processors) is responsive to a client query and uses an initial search queue.

At column 16, lines 9-21 of Kitain, there is mention of two queries per a “report query form” of relational database 11 accessed by a common gateway interface (CGI) described at column 13 as an interface between the web server 4 program and other programs. This passage may comprise a suggestion of an initial query “for a list of all contributors” followed by a second query “for a list of all industries” but not as amended: “an initial search queue of at least said client query” where said client query is “for said specific data of said database” having “a

database index . . . comprising nodes of a database tree” and so on. For all these additional reasons, applicants urge that claim 46 as amended patentably distinguishes over Merkey/Kitain.

Claim 48 as amended

Claim 48 as amended is allowable for all the reasons claim 42 is allowable and further incorporates the limitation the root host processor being responsive to a client query and selecting a host processor to receive search query information. The Examiner refers the applicant to Merkey Figure 6 and Kitain col. 16 lines 44-47. Kitain has a search engine but no such limitation as recited. As urged in relation to claim 46, Merkey and Kitain even if combined fail to teach this selection limitation. Consequently, claim 48 is allowable over a Merkey/Kitain combination. Kitain tries each processor to accept a query, and Merkey collects available thread count from many via structure 67 and maintains a local thread queue at each processor rather than broadcasting capacity and load from many to many and maintaining a memory at multiple processors about many as recited. Merkey and Kitain represent apples and oranges that cannot be mixed.

Claim 66 as amended

At pages 15-17 of the Office Action issued June 27, 2007, the Examiner suggests that Merkey Figure 2 and 5 show a plurality of host processors comprising at least one root host responsive to a client query. Figure 2, in fact, shows P1 to P4 intercoupled to modules M1 to M4 via crossbar switch. Figure 5 shows P1 to P4 associated with structure 67 and global dispatch queue 68 with more threads 66 shown at local queue 64 for P4 than P1 – again Merkey using thread count as a measure of load/capacity at each processor. To the contrary, Merkey provides no discussion of a “client query for said specific data of said database” as recited or “each host processor bringing its search queue of client queries into balance with another host processor according to a time constant” as recited. Again, Merkey is concerned with a centralized thread queue 67, not a “search queue of client queries” as recited, and a “processor in question” checking the central queue and then local queues to find a processor thread. Claim 66 as amended further includes limitations that the balancing process involves “stochastic selection” to

modify the step of bringing a search queue into balance (see support at least at page 48, ll. 7-18). There is no specific suggestion to use stochastic selection in Merkey, or by Kitain to use stochastic selection in a non-search environment such as Merkey, to determine the recipient host processor of an exchanged search request between host processors as recited. The examiner cites to Kitain, col. 16, lines 44-47, “semi-random selection of servers.” “This means that the order that servers are tried is not always the same.” Reading before those words at col. 16, we learn that the process of satisfying a “non-text matching query” will iteratively try one type of server, then another and so on until a server satisfies the query or all servers are found to be down. Merkey fails to provide any disclosure or suggestion of an “exchanged search request” as recited or at least, “each host processor bringing its search queue of client queries into balance with another host processor according to a time constant . . .” Kitain fails to make up for Merkey’s deficiencies. Also, the Examiner is respectfully requested to consider the limitation, “said balancing including exchanging unprocessed search requests with a recipient host processor responsive to a stochastic selection process to determine the recipient host processor of an exchanged search request between host processors, said balancing thereby minimizing the time required to respond to client queries for retrieval of responsive data from said database.” Reconsideration and allowance of claim 66 as amended is respectfully requested.

Claim 68 as amended

Claim 68 has been discussed above as containing all the limitations of claim 66 and the recited limitation related to “bringing . . . into balance further comprises exchanging a block of search requests between host processors, thereby minimizing a time to respond to client queries by said exchanging of said block of search requests.” No such concept is disclosed or suggested by Merkey or Kitain.

Claim 72 as amended

Claim 72 stands rejected on the grounds that Merkey Figures 2, 5, 7 and 9 show the recited elements along with the passage at column 9 lines 30-40. Kitain is added because the examiner admits that the search, storage and retrieval of data, database index, among other

elements, are not shown by Merkey. The examiner points to column 6 lines 12-27 and column 16 lines 44-47. The examiner states at page 18 of the office action that a motivation for combining Kitain and Merkey is an efficient search engine. The examiner, using improper hindsight, suggests that this “efficient search engine being used to obtain the search results and allows more than one search to be conducted in parallel . . . Furthermore, with the global dispatch queue the load balancing of the processors would make the processing much more efficient.” This is clear evidence that the examiner has reached an improper hindsight conclusion and of what combination? Claim 72 as amended requires “at least two host processors . . . maintaining information on said plurality of said available host processors and on their processor capacity and load including information of a search queue length of client queries” not taught by either Merkey or Kitain.

The Examiner appears to be suggesting taking a thread control structure 67 and global dispatch queue 68 from Merkey Figure 5 and apply the thread queue structure 67 along with local queues 64 in Kitain Figure 1. The rejection is stated the other way around as Merkey in view of Kitain. Consequently, the Examiner has failed to suggest even the possibility of such a combination or how the elements of claim 72 as amended are found in Kitain as a primary reference. Even assuming the other way around, there is no search engine in Merkey and no “at least two processors” limitation as copied above. Applicants therefore respectfully submit the rejection of claim 72 is improper, and request that it be withdrawn. Again, neither Merkey nor Kitain teach the limitation of claim 72 as amended above or “each host processor (comprising at least two) broadcasting its capacity and load information to other host processors” or the newly recited addition “and bringing each search queue of client queries into balance with another host processor according to a time constant in response to receipt of broadcast capacity and load information, said bring into balance comprising equalizing average waiting times for service and computation between said search engines.” To the contrary, Kitain has refers to one search engine at a time when satisfying a non-text or text matching search query practicing a semi-random selection process. Also, neither Merkey nor Kitain teach “stochastic selection of host processors to determine a recipient host processor for an exchanged search request for searching said database” as now recited.

Dependent claims of special note

Claim 49/42 has been amended to recite “the receipt of broadcast search queue length load and gathered processor capacity information.” Claim 58/46 has been amended to recite the balancing including exchanging unprocessed search requests with a recipient host processor responsive to a stochastic selection process. Claim 68/66 now recites regarding a block of search requests: “thereby minimizing a time required to respond to client queries of said exchanging of said block of search requests.” The dependency of claim 69/68 has been corrected. Claims 73/72 and 74/72 directed to failure and addition of host processors respectively have been respectively amended to identify further limitations.

The examiner fails to reject claims 75 or 76 except via a reference at page 4 of the Office Action which suggests that these claims are rejected under the same rationale given for claim 42 yet they contain new limitations shared memory and distributed memory between host processors or among each processor as recited. The Examiner has failed to indicate where in Merkey or Kitain these additional limitations are found. Moreover, applicants respectfully request reconsideration and allowance of all other remaining dependent claims not discussed above which add limitations to the current set of allowable independent claims.

CONCLUSION

All rejections of independent claims 42, 44, 45, 46, 48, 66 and 72 having been addressed, applicant respectfully submits that the instant application is in condition for allowance, and respectfully reconsideration of all pending claims and solicits prompt notification of the same. Applicants have also called the Examiners' attention to specific limitations of dependent claims which have not previously been examined in context or as amended. However, if for any reason the Examiner believes the application is not in condition for allowance or there are any questions, the Examiner is requested to contact the undersigned at (202) 624-7325.

Respectfully submitted,

POWELL GOLDSTEIN LLP

Dated this 19th day of September, 2007

By: /Thomas H. Jackson/
Thomas H. Jackson, Registration No. 29,808
901 New York Avenue, N.W.
Washington, D.C. 20001-4413
Tel: (202) 624-7325
Fax: (202) 624-7222

THJ/mmd